**Yêu cầu:**

**- Tìm hiểu về giải thuật TCP CUBIC, TCP Compound và TCP BBR**

**- So sánh 3 giải thuật trên.**

**- Cho biết ưu, khuyết của 3 giải thuật này.**

**Lưu ý: Tập trung vào các nội dung sau:**

**1. Lý do ra đời của giải thuật ( vấn đề các giải thuật cũ, giải pháp...)**

**2. Giai đoạn khi mới khởi tạo kết nối TCP thì hoạt động kiểm soát tắc nghẽn thế nào.**

**3. Khi gặp tắc nghẽn thì thế nào ?**

**4. Giai đoạn sau tắc nghẽn diễn ra thế nào ?**

**1. Lý do ra đời của TCP BBR**BBR là một dự án của Google được nghiên cứu khi các kĩ sư nhận ra vấn đề trong các thiết kế mạng trong hệ thống loss-based congestion control trong hệ thống hiện tại. Câu chuyện diễn ra vào năm 2011, khi mà nhiều kĩ sư làm việc trong lĩnh vực mạng viễn thông nhận ra việc quá tải buffer cũng như chậm tiến độ đường truyền thường xuyên diễn ra ở những đoạn cuối cùng của hệ thống internet lúc bấy giờ. Vấn đề này đã được định nghĩa là bufferbloat. Các gói tin HTTP được gửi đi làm cho hệ thống chậm đi hơn cả những gói tin HTTP 1.1 ( là một phiên bản cũ ) khiến cho việc gửi lại đi rất nhiều kết nối TCP. Năm 2013, các kĩ sư làm việc ở google nhận ra khi sử dụng mạng cục bộ trong hệ thống với tốc độ cực kì cao nhưng lại gửi những gói tin đi cực chậm vì mạng được cấu hình bằng shallow buffer switches và khi truyền một lượng lớn gói tin và thông tin trong một vùng rộng hơn dẫn đến một vài vấn đề nghiêm trọng. Tất cả những vấn đề gặp phải phía trên đều có nguyên nhân cốt lõi là từ hệ thống loss-based congestion control ( eg. Reno, Cubic, etc) là việc **đóng gói lẻ một gói tin** làm cho hệ thống nhận diện tắt nghẽn hoạt động kém hơn rất nhiều, một vấn đề nữa là có thể gặp phải packet loss trước khi có thể kiểm soát được tắt nghẽn

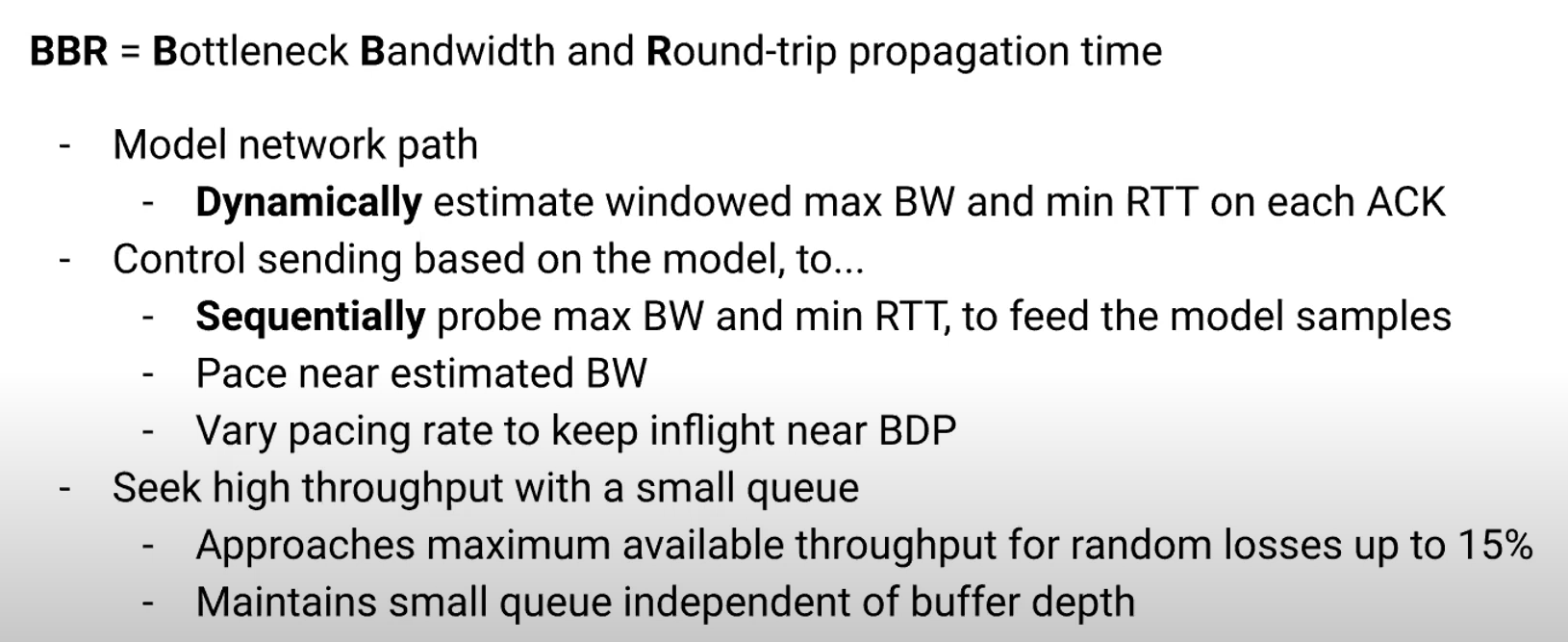
——————————————————————————————————————————

**2. Hoạt động kiểm soát tắc nghẽn khi khởi tạo kết nối**

* Ngay từ khi khởi tạo kết nối, BBR bắt đầu đo lường băng thông tắc nghẽn (BtlBw) và thời gian vòng tròn (RTprop). Những thông số này sẽ được sử dụng để điều chỉnh tốc độ gửi dữ liệu sau khi kết nối được thiết lập.
* Sau khi nhận được gói tin ACK cuối cùng, BBR sẽ sử dụng các giá trị băng thông và RTT đã đo được để tối ưu hóa tốc độ gửi dữ liệu, nhằm đạt được hiệu suất tốt nhất mà không gây ra tắc nghẽn.
* Trong giai đoạn khởi động, BBR sẽ nhanh chóng tăng tốc độ gửi dữ liệu để tìm ra băng thông tối đa có thể sử dụng trước khi chuyển sang các giai đoạn thăm dò băng thông khác.

——————————————————————————————————————————

**3. Hoạt động khi gặp tắc nghẽn**



## Cách BBR Kiểm Tra Tắc Nghẽn

1. Đo Lường Băng Thông Tắc Nghẽn (BtlBw):
   * BBR liên tục ước lượng băng thông tắc nghẽn bằng cách theo dõi lượng dữ liệu được xác nhận (ACK) trong một khoảng thời gian nhất định. Khi băng thông đạt đến mức tối đa mà không còn tăng trưởng, BBR sẽ chuyển sang giai đoạn khác để xử lý tắc nghẽn.
2. Thời Gian Vòng Tròn (RTT):
   * BBR ghi lại thời gian rời khỏi mỗi gói tin để tính toán RTT. Thông qua việc theo dõi RTT, BBR có thể điều chỉnh tốc độ gửi dữ liệu và xác định xem có xảy ra tắc nghẽn hay không. Nếu RTT tăng lên, điều này có thể chỉ ra rằng có sự tắc nghẽn trong mạng.
3. Giai Đoạn Khởi Động (STARTUP):
   * Trong giai đoạn này, BBR tăng tốc độ gửi dữ liệu nhanh chóng để tìm ra băng thông tối đa. Khi tốc độ gửi không còn tăng nữa, BBR sẽ chuyển sang giai đoạn DRAIN.
4. Giai Đoạn DRAIN:
   * Sau khi ước lượng được băng thông, BBR sẽ giảm tốc độ gửi để làm sạch hàng đợi đã tạo ra trong quá trình khởi động.
5. Giai Đoạn Thăm Dò Băng Thông (PROBE\_BW):
   * BBR thử nghiệm băng thông bằng cách tăng và giảm tốc độ gửi dữ liệu để tìm kiếm khả năng tăng băng thông mà không tạo ra hàng đợi quá lớn.
6. Giai Đoạn Thăm Dò RTT (PROBE\_RTT):
   * Nếu không có cập nhật RTT trong một khoảng thời gian nhất định, BBR sẽ giảm tốc độ gửi xuống mức tối thiểu để đo lại thời gian truyền đi mà không bị ảnh hưởng bởi hàng đợi.

1. Đo lường băng thông và thời gian truyền:

* BBR liên tục đo lường băng thông tối đa mà mạng có thể xử lý và thời gian vòng đi (RTT). Khi phát hiện tắc nghẽn, nó sẽ điều chỉnh tốc độ gửi dựa trên các thông số này, giúp tránh việc gửi quá nhiều dữ liệu vào mạng trong thời gian tắc nghẽn

2. Điều chỉnh kích thước cửa sổ truyền tải:

* BBR điều chỉnh kích thước cửa sổ truyền tải (Congestion Window - cwnd) một cách linh hoạt. Khi phát hiện dấu hiệu tắc nghẽn, nó sẽ giảm kích thước cwnd để hạn chế lượng dữ liệu gửi đi, từ đó giảm thiểu khả năng mất gói và cải thiện hiệu suất tổng thể [1](https://quantrimang.com/lang-cong-nghe/google-tang-toc-internet-nhu-the-nao-139866)[2](https://niithanoi.vn/chi-tiet-tin/1318/google-da-lam-cach-nao-de-tang-toc-internet.html).

3. Không chờ đợi mất gói:

* Khác với các thuật toán kiểm soát tắc nghẽn truyền thống, BBR không chờ đến khi mất gói mới điều chỉnh tốc độ gửi. Thay vào đó, nó chủ động điều chỉnh tốc độ dựa trên các phép đo băng thông và RTT, giúp ngăn chặn tắc nghẽn trước khi xảy ra [1](https://quantrimang.com/lang-cong-nghe/google-tang-toc-internet-nhu-the-nao-139866)[2](https://niithanoi.vn/chi-tiet-tin/1318/google-da-lam-cach-nao-de-tang-toc-internet.html).

4. Tối ưu hóa tốc độ gửi:

* BBR cố gắng duy trì tốc độ gửi ở mức mà mạng có thể xử lý được mà không gây ra tình trạng tắc nghẽn. Nó tính toán lượng dữ liệu tối đa có thể gửi mà không làm mạng bị nghẽn, từ đó điều chỉnh tốc độ gửi để phù hợp với khả năng của mạng

## Kết Luận

BBR sử dụng các phương pháp đo lường băng thông và RTT một cách liên tục để điều chỉnh tốc độ gửi dữ liệu, nhằm phát hiện và xử lý tắc nghẽn hiệu quả hơn so với các thuật toán truyền thống như TCP CUBIC. Điều này giúp cải thiện cả hiệu suất truyền tải và độ trễ trong mạng.

——————————————————————————————————————————

**4. Ứng xử sau tắc nghẽn**

Các Bước Sau Khi Giải Quyết Tắc Nghẽn

1. Quay Lại Giai Đoạn Thăm Dò Băng Thông (PROBE\_BW):
   * BBR sẽ chuyển sang giai đoạn thăm dò băng thông để kiểm tra khả năng tăng băng thông. Trong giai đoạn này, BBR sẽ điều chỉnh tốc độ gửi dữ liệu để tìm kiếm băng thông tối đa mà không gây ra tắc nghẽn. Nó sẽ tăng cường tốc độ gửi dữ liệu và theo dõi phản hồi từ mạng để xác định băng thông hiện tại.
2. Cập Nhật Thời Gian Vòng Tròn (RTT):
   * BBR liên tục theo dõi và cập nhật thời gian vòng tròn (RTT) để phản ánh chính xác độ trễ của mạng. Nếu RTT giảm, điều này có thể cho thấy rằng mạng đang hoạt động tốt hơn và có thể cho phép tăng tốc độ gửi dữ liệu.
3. Điều Chỉnh Tốc Độ Gửi Dữ Liệu:
   * Dựa trên các ước lượng về băng thông tắc nghẽn (BtlBw) và thời gian vòng tròn (RTprop), BBR sẽ điều chỉnh tốc độ gửi dữ liệu để duy trì hiệu suất tối ưu. Mục tiêu là giữ cho lượng dữ liệu đang bay trong mạng (in-flight data) gần với sản phẩm băng thông-độ trễ (BDP), nhằm tối ưu hóa throughput và giảm độ trễ.
4. Giai Đoạn Thăm Dò RTT (PROBE\_RTT):
   * Nếu không có cập nhật nào về RTT trong một khoảng thời gian nhất định, BBR sẽ vào giai đoạn thăm dò RTT, nơi nó giảm số lượng gói tin đang bay xuống mức tối thiểu để đo lại thời gian vòng tròn mà không bị ảnh hưởng bởi hàng đợi.
5. Duy Trì Hiệu Suất:
   * BBR tiếp tục theo dõi và điều chỉnh tốc độ gửi dữ liệu dựa trên các điều kiện mạng hiện tại, nhằm đảm bảo rằng throughput được duy trì ở mức cao nhất có thể mà không gây ra tắc nghẽn.

——————————————————————————————————————————

**5. Ưu điểm và nhược điểm của TCP BBR**

| Ưu điểm | Nhược điểm |
| --- | --- |
| - BBR cung cấp khả năng tăng thông | Trong môi trường có nhiều luồng TCP khác nhau (như CUBIC hoặc Reno), BBR có thể chiếm ưu thế quá mức |
| BBR sử dụng phương pháp giúp tránh hiện tượng bufferbloat, nơi mà hàng đợi quá lớn gây ra độ trễ. | Bất lợi khi nhiều phiên BBR hoạt động đồng thời |
| BBR có khả năng phản ứng nhanh với các thay đổi trong điều kiện mạng | - không phản ứng nhanh chóng như một số thuật toán kiểm soát tắc nghẽn dựa trên mất gói khác. |
| BBR hoạt động hiệu quả trong các môi trường mạng có tốc độ cao và bộ đệm nhỏ, | Việc triển khai BBR có thể yêu cầu thay đổi cấu hình mạng hiện tại để tối ưu hóa hiệu suất, điều này có thể gây khó khăn cho một số nhà cung cấp dịch vụ |
| - BBR yêu cầu ít bộ đệm hơn trong mạng, điều này có thể dẫn đến chi phí thấp hơn cho hạ tầng mạng mà vẫn đảm bảo hiệu suất cao | - Phiên bản đầu tiên của BBR (BBR v1) đã chỉ ra một số hạn chế, dẫn đến việc phát triển phiên bản cải tiến (BBR v2) để khắc phục những vấn đề này, nhưng vẫn cần tiếp tục theo dõi và cải thiện |

——————————————————————————————————————————

**6. Kết luận**

BBR không chỉ đơn thuần là giải quyết tắc nghẽn mà còn tiếp tục tối ưu hóa quá trình truyền tải dữ liệu bằng cách thăm dò băng thông và điều chỉnh tốc độ gửi dựa trên các đo lường thực tế về băng thông và độ trễ. Điều này giúp cải thiện hiệu suất tổng thể của kết nối mạng. BBR là một cải tiến đáng kể trong kiểm soát tắc nghẽn TCP, mang lại nhiều lợi ích về hiệu suất truyền tải dữ liệu. Tuy nhiên, nó cũng đi kèm với một số thách thức cần được giải quyết để đảm bảo sự công bằng và ổn định trong môi trường mạng đa dạng hiện nay.

**References**

1.<https://aws.amazon.com/vi/blogs/networking-and-content-delivery/tcp-bbr-congestion-control-with-amazon-cloudfront/>  
2. <https://wiki.geant.org/pages/releaseview.action?pageId=121340614>  
3. <https://www.youtube.com/watch?v=VIX45zMMZG8&t=1208s>  
4. <https://www.ietf.org/archive/id/draft-cardwell-iccrg-bbr-congestion-control-01.html>  
5. <https://labs.apnic.net/presentations/store/2019-09-05-bbr.pdf>

7. <https://www.perplexity.ai/search/history-of-tcb-bbr-tknoeTSBT7.BM9cG0QwY0g>